Debian Linux on an Averatec 3200 Series laptop

(version 1.16)

Gene Cooperman

I am using an Averatec 3220 notebook (3200 series). I also installed Debian Sarge on a 3270. Since then, I upgraded to Etch. First, I describe the current issues, followed by a listing of what I did to configure each component of the laptop.

As of summer, 2007, the Averatec battery could not be charged. (The battery light would blink very intermittently.) This site gives instructions for disassembling and re-soldering. This other site gives information for the Averatec 3200.

*WARNING: As of my current Linux 2.6.17 kernel, this problem has gone away. I will keep this message here a little longer for those with older kernels. After I had upgraded to the 2.6.12 kernel, the Synaptics touchpad now occasionally freezes. (Don't know if this happens with later kernels.) Tracing comments on the web, I found I had to add **append="i8042.nomux"** to a stanza in lilo.conf, due to a bug in the i8042 device. If using grub instead of lilo, then add **i8042.nomux** to the end of the kernel line.*

WARNING: The vendor of the Smartlink modem appears no longer to support a Linux driver for Linux kernel 2.6.12 or later. See the modem section below for details of how to use either the open source ALSA mode modem (built into some kernels) with the kernel module, snd_via82xx_modem. This might not allow you to use sound and modem at the same time. An alternative (with some binary-only parts) described in the modem section uses the Smart Link modem, and seems to provide a more reliable connection.

(I also recommend the Averatec notebook, if an economical thin and light laptop is a priority (but see caveat). The Averatec is thin and light, 512 MB, integrated wireless, and under $1,000. The 3270 model includes a DVD burner. However, as one would expect, there are compromises. The battery initialy had 1-1/2 hours capacity, but after 1-1/2 years, the battery had less than 1/2 hour capacity. Also, after 2 years, the DVD/CD-writer is no longer able to write, although it still reads. A second Averatec notebook started overheating with 2 or 3 hours of use, after one and a half years.) Below are my experiences in configuring Debian on the Averatec.

These notes can easily adapted to Debian on another computer. In fact, I wrote these notes down carefully, to use on other computers or helping friends. Additions and corrections are welcome: gene@ccs.neu.edu . As usual, there are no guarantees. Use these suggestions at your own risk.

One general piece of advice, is to first run Knoppix or Kanotix. Under Knoppix, look at /etc/sysconfig (Does this also work under Kanotix?). That directory describes the hardware auto-detected by Knoppix, along with appropriate device drivers. `lspci` and `usbview` are other useful tools for detecting devices.

One other piece of advice: use "aptitude" to manage the Debian packages. If you play with it a little, you will find it very powerful. If you fail to configure a package during the first installation, don't forget that `dpkg-reconfigure is always available.`

- Partitioning the hard disk
- Debian Reference (after initial install)
- Video
- Sound
- Multimedia Apps
- Internationalization/Accented Characters
- Ethernet
- Wireless
- Modem

- CD-RW/DVD (writing CD's and DVD's)
- RAM disk
- Touchpad and Keyboard
- Power Management and Adjust CPU Speed
- Hibernate/Suspend
- USB (mice, keyboards, flash drives, digital camera, USB videocam/webcam, etc.)
- Reconfiguring the kernel
- Installing a kernel module
- Recovering when Debian kernel doesn't boot
- Minor bugs that I haven't fixed

One minor complaint is that there appears to be no hardware reset button on the Averatec 3200. If you like to hack around like I do, you will eventually get the kernel to freeze. The Linux equivalent of CTRL-ALT-DEL is Alt-SysReq-KEY, where KEY is b (for reboot), k (for kill programs on current virtual console) and others documented in sysrq.txt in the Linux kernel documentation. A good pattern for emergency shutdown is:

1. Alt-SysRq-S: Emergency Sync (write everything on hard disk)
2. Alt-SysRq-U: Unmount all harddisks
3. Alt-SysRq-B: Reboot system immediately

(Try `locate sysrq.txt` or google for it on the web.) If this does not work, unplug the battery to force the laptop to power down.

For some similar reports on the Averatec 3200 series, see:

- Linux on Laptops -- Averatec (compilation of reports on Averatec)
- Mepis Linux on Averatec AV3250HX-01 is another excellent web page, with additional details on installing multimedia apps. Mepis is a Debian-based distro.

I am not a Linux guru. Here are notes on what I did to get all the hardware features of the Averatec working under Debian. I offer the notes in case they may help you. But naturally, I provide no guarantees. Use these comments at your own risk. I installed Debian 3.0 (Woody) and then upgraded to Debian "unstable". (I'd probably just use "testing", if I did it again.)

I've also installed Sarge (testing, as of 4/05) on an Averatec 3270.

I also tested under Knoppix 3.6, and most features work. Note that Knoppix can be installed directly to your disk.
EXCEPTION: For wireless, ndiswrapper is provided by Knoppix, but a Windows driver is still needed (see below).
EXCEPTION: the modem doesn't work; The modem is SmartLink. I got the modem working with the free driver for SmartLink modem. (Although see comment at top about 2.6.12 kernel.)

If you want the easy way out, just get a Knoppix, and install it to hard disk. Then copy the Windows drivers below for wireless and install it using ndiswrapper. Also get the free Linux driver for the built-in SmartLink modem. However, your primary upgrade path will then be to get newer versions of Knoppix, since Knoppix (as of 3.6) uses a mix of testing and unstable packages. Upgrading Debian packages the normal way requires you to choose "testing" or "unstable".

Some of the comments below may also be helpful to configure a different distribution other than Debian or Knoppix.

```
================================================
I use Debian, but I also run the Knoppix LiveCD for hints about configuration.
I added 'battery' to /etc/modules, and  /proc/acpi/battery/BAT0/state
  now reports the battery level.  (Perhaps this is standard with current
  Debian installations.)
I might have tried adding 'via-rhine' to /etc/modules for Ethernet, but
  I didn't need it.
Knoppix includes a command `acpi -b' with a nicer summary of the information.
```

==========

Partitioning the hard disk:

Today, I would use a *recent* version of qtparted on a LiveCD. Qtparted is a front-end for ntfsresize. (Note that older versions, for example older than version 1.11.2, may have problems, and are not recommended. Knoppix 4.02 uses a qtparted with ntfsresize version 1.11.2.) (Originally, I had used the setup phase of an old Mandrake 8.2 CD to resize the Windows NTFS partition and create partitions for Linux.) My notebook was new, and so Windows was not fragmented. Otherwise, I would defragment first as insurance. Partition Magic, qtparted or some other software may also work. Personally, I sometimes prefer parted to the qtparted GUI front-end, since parted has a bit more documentation.

If you use Grub as your bootloader (the most common choice, today), you will have no problem. I used the Lilo bootloader. I also had an NTFS partition (Windows XP Home), a FAT32 partition (readable/writeable from Windows and Linux), and the usual Linux partitions. My "other" stanza in /etc/lilo.conf refers to the NTFS partition. Recent versions of Lilo have automatic changes. Lilo will say something like "ASSUMING AUTOMATIC CHANGE-RULES" when you call lilo. The result was that each time I booted under Windows, Lilo set the FAT32 partition to a hidden partition, and it was no longer visible in Windows. To fix this, I modified my "other" stanza as follows:

```
other=/dev/hda1
  label="WinNT(hda1)"
    change
```

The "change" command says to use our own empty change-rules and not the automatic change-rules. The automatic change-rules were the ones making the inactive dos partition hidden (on the theory that it would confuse Microsoft Windows?).

I then ran "lilo", and finally "parted" (or qtparted). Using parted, I found the FAT32 partition with the "hidden" flag as follows: (parted) print [This showed partition 2 to have the hidden flag.] (parted) set 2 hidden off [The number 2 is partition 2.] (parted) print [Check partition 2 now has no hidden flag.] (parted) quit

---

Debian Reference (after initial install):

After initial installation of Debian, *do* get the Debian package `debian-reference-en`. After that, the menu item "Applications->Debian Menu->Help->Debian Reference" will appear. It is also available at http://www.debian.org/doc/manuals/reference/, or the latest development version is at http://www.debian.org/doc/manuals/reference/. It's missing chapters on such things as multimedia, sound, USB flash drive, USB camera, but what it has is concise, to-the-point, and very helpful. A good source for the rest (multimedia, sound, USB, etc.) is Debian GNU/Linux Desktop Survival Guide.

For networking (especially for gateways, routers), I like this Linux Proxy server tutorial (firewall) with associated background material (proxy/nat, networking, modems, etc.). Chapter 10 of the "Debian Reference" also has some information.

Video:

Conceptually, there is a vesa graphics mode that all modern hardware can produce. In addition, any accelerated graphics mode will require a special driver (outside the kernel) from XFree86 or X.org. Above the hardware, there is an implementation of the X-windows system: usually from XFree86 or X.org. Above the X Windows system is a desktop environment (often Gnome, KDE, Enlightenment, etc.). Above the desktop is a window manager (typically, the same brand as the desktop, but there are options to mix and match). It's also possible to run some window managers directly on top of X Windows.

You can force the X Windows system to exit (and possibly restart) by typing <ctrl><alt><backspace>. You can get to additional text-oriented consoles (no graphics) by typing <ctrl><alt><F1>, <ctrl><alt><F2>, etc. Typically, the grpahics-oriented console is at <ctrl><alt><F7>.

I think I had to tweak the XF86config-4, but I don't remember the details. I believe I just merged the

Knoppix version of /etc/X11/XF86config-4 with the Debian one, until I had what I wanted. You're welcome to copy my own configuration: http://www.ccs.neu.edu/home/gene/XF86Config-4.averatec You can also find /etc/sysconfig/XF86Config-4 while you're in Knoppix. Once video is working, you may want to look at the Multimedia section.

Sound:

Conceptually, the layers of sound-related software are: first, the sound device driver as a kernel module; second, the ALSA software API (which can emulate the older OSS software API if it is configured so in the kernel); and third, a sound server with mixer capabilities (mixing sound sources; aRts for KDE, ESD for Enlightenment, and configurable for Gnome --- ESD, none, etc.). OSS uses /dev/audio, ALSA uses /dev/dsp, and sound serers typically use /dev/mixer. The sound hardware is locked by either a single sound application or else by a sound server that allows multiple applications to use sound.

The sound server is not required by sound applications, and simply provides additional facilities (mixing, transport of sound across the network, etc.) Under Gnome, you can choose your sound server via "Desktop>Multimedia Systems Selector" and selecting "Default Sink" and "Default Source". (Choosing Alsa or OSS essentially means no sound server.) You can turn on or off use of the sound server via "Desktop->Preferences->Sound", and checking "Enable sound server startup".

As of Sarge (Debian 3.1) and Etch, the Debian kernels seem to include most sound device drivers, but only as modules. Etch seems to correctly add the necessary via82cxxx_audio module to /etc/modules. For the older Sarge, I had to add it.

If in doubt on the correct /etc/modules, note that Knoppix/Kanotix automatically fills in /etc/modules and provides OSS (the old sound API) by default **(true for Kanotix?)**. Other ways to guess the correct driver are `lspci -v` and `cat /proc/bus/pci/devices`.

Alternatively, if you recompile the kernel to include the device drivers for your computer as built-in, you do not need to add to /etc/modules. See the kernel section of this document.

The kernel supports sound via ALSA (Advanced Linux Sound Architecture), or the older OSS (Open Sound System, DEPRECATED). The older OSS supports sound through the devices /dev/audio (raw sampled files), /dev/dsp (OSS compatible PCM or digital audio), and /dev/mixer. The newer ALSA kernel modules are always named snd-XXX. Examples are snd-via82xx (via82xxx sound chip), snd-pcm-oss (OSS emulation to support /dev/dsp), snd-mixer-oss (OSS emulation to support /dev/mixer), and a rich variety of other devices.

If you want to simply use OSS directly, you will want the module `via82cxxx_audio'. In the kernel source directory within `make menuconfig' (libncurses5 and libncurses5-dev packages needed), select

```
Device Drivers->Sound->Open Sound System:
                    VIA 82C686 Audio Codec (NEW)
```

Note that xmms and aumix use the older OSS interface. (Actually, xmms uses the older OSS by default, but if you access its "configure" (click "O" for options, and then "preferences"), you can set it to use ALSA instead of OSS. Select the ALSA output plugin. If you want to play iPod files, make sure you choose something for m4a for the input plugin. xmms-mp4 seems to work, available at: http://www.debian-multimedia.org/ (multimedia section).) OSS works through /dev/audio. Try `cat ANY_LARGE_FILE > /dev/audio' to test the OSS interface.

Newer programs like alsamixer use only the newer Alsa interface. First, make sure you have Alsa installed. In Debian, I use the packages: alsa-base, alsa-utils, and alsa-oss. If gnome-volume-control, the sound volume applet in Gnome, is permanently stuck on 0 volume, this is an indication that you don't have Alsa installed. Once Alsa is installed, to enable it, make sure to run:

alsaconf

With luck, it will detect your card, and configure the necessary system files. If that doesn't work for you, read on.

Make sure you have Alsa support configured into your kernel. In the kernel source directory, under `make menuconfig' (libncurses5 and libncurses5-dev packages needed), I selected:

```
Device Drivers->Sound->Alsa:
   Alsa; (MIDI) Sequencer; OSS Mixer API;
                        OSS PCM (digital) audio; OSS sequencer API
Device Drivers->Sound->Alsa->PCI:
   Via82c686A/B, 8233 SouthBridge
```

I chose to compile the options built-in. Without that, you have to load the Alsa modules in /etc/modules. I included:

```
sound
snd-via82xx   (snd_via82xx)
snd-pcm-oss   (snd_pcm_oss)
snd-mixer-oss (snd_mixer_oss)
[ OTHERS? ]
```

For Alsa, check the /proc/asound/ directory to verify that the kernel has set up your modules (e.g. /proc/asound/cards to see if the kernel recognizes your sound card). Check also that amixer or gnome-alsamixer works. (You may need to either try them as root, or else place the user in the group audio: adduser USER audio.)

Next, in theory, do:

1. alsaconf [and accept all of its suggestions]
2. aumix [and adjust defaults as desired]
3. alsactl store [and save defaults, so they're available next session]

You may also need to run "Applications->Sound&Video->Volume Control" (gnome-volume-control), and "gnome-alsamixer". Try *both*. If alsa sets the volume to zero at the lower level, or if gnome-volume-control sets the volume to zero at the higher level within Gnome, then you will not hear sound. Also, in aumixer, did you raise sound above zero for CD, Video, PCM, line, etc.? For example, you won't hear sound in video if you don't enable Video (and possibly also PCM).

After this, xmms/aumix (using OSS interface) and alsaplayer/alsamixer (using ALSA interface) both work. The Gnome applications don't have man pages, and I haven't found a good HOWTO on setting up Gnome. I find the Gnome help documentation less than useful. It seems primarily to be of the form:

"The File->Open menu item is used to open a file." (and when they go all out, they will show you a picture of what the "File->Open" menu item looks like.)

On top of ALSA, Gnome provides sound servers. If you get noise when Gnome starts, try disabling the sound server (see "Enable sound server startup" above). If you want multiple applications to be able to play sound simultaneously, then try enabling it.

With Alsa working, OSS compatibility should also work, assuming that was selected in kernel configuration. Check OSS as above.

Finally, under early kernel 2.6.12, I used ESD (Enlightened Sound Daemon) when gnome started, I got noise. You may not see this under later kernels. I traced it down to the process (created when I login under Gnome's gdm):

```
/usr/bin/esd -nobeeps    [the Enlightened Sound Daemon of Gnome]
```

Killing esd stopped the noise, and it did not return when esd is reexecuted. You can also disable the esd from the Gnome control menu (see above).

If you want to be able to record any device using PCM (/dev/dsp), use the vsound package in Debian. For example:

vsound -f output.wav realplay input.rm

The KDE package, realrekord, exists as a graphic front-end for vsound. Another possibility is

"Applications->Sound&Video->Sound Recorder" (gnome-sound-recorder).

Multimedia Apps:

For painlessly installing multimedia apps apps, I added a line:
`deb http://www.debian-multimedia.org testing main`
to `/etc/apt/sources.list` . I could then `apt-get install mplayer-k7`, etc. See http://debian.video.free.fr/ for further information, or probably now simply http://www.debian-multimedia.org/. Note also, the possibility to add: `deb http://www.debian-multimedia.org experimental main`
to get such packages as *dvdrip*.

PyMusique (or alt) is available as an interface to the Apple iTunes Music Store.

Some other multimedia apps to look into are: Scribus (desktop publishing); Blender (tools for 3D creation); Banshee (iTunes replacement and more); gtkpod (alt iTunes replacement to manage iPod; see below for its configuration); Amarok (alternative to gtkpod, but not as reliable?); Audacity (audio editor and recorder); Ardour (multichannel digital audio workstation); Jamin (JACK Audio Connection Kit (JACK) Audio Mastering interface); toolame (for creating mp3's (layer 2, to avoid patent issues of lame) -- a Debian package); hugin (easily create panoramas from overlapping pictures); Kino (non-linear DV editor for capture, VTR control, and recording back to the camera); and dvdbackup (extracts DVD in format for burning later, easy to use --- a Debian package)

Separately, MediaInlinux is a multimedia LiveCd with most of the above packages. 64 Studio for digital content creation is a Debian Pure 64 offshoot for AMD-64 and EMT64 only, but it ibut it is still only in testing.

If the microphone doesn't record loudly enough, note that Alsamixergui has a setting, "Mic Boost (+20)". This has no volume, but the speaker icons can be turned on and off. Turn it on if your microphone volume is not loud enough, and don't forget "Mic Select"..

As for configuring Debian/Gnome to work with iPod, I added: In Gnome: Desktop -> Removable Drives and Media Preferences:
Multimedia: iPod: Check "Sync files when connected"
and specify: Command: gtkpod
banshee is now available as a newer software.

In theory, it should then be enough to go to:
Edit->Preferences: General: Import:
and set iPod Mount Point: /media/ipod
and also check "Handle mounting/unmounting of iPod drive"
However, as of gtpkod version 0.99.2 (I actually use gtkpod-aac from debian-multimedia.org, which should properly play .m4a files), on quitting from gtkpod, gtkpod calls "umount" instead of "eject". As a result, the disk is unmounted, but the iPod is unaware of this, and continues to indicate "Do not disconnect" on its screen.

The fix is to leave "Handle mounting/unmounting of iPod drive" *unchecked*, and instead add a file
`~/.gtkpod/gtkpod.out` : `#!/bin/bash`
`/usr/bin/sudo /usr/bin/eject /media/ipod`
and to also add to `/etc/sudoers` a line:
`ALL ALL= NOPASSWD: /usr/bin/eject /media/ipod`

Finally, in using gtkpod, note that you can select "iPod" and then drag the text for albums or tracks onto "Local" in order to copy from iPod to disk. Then click "Sync" before quitting. The reverse should work in the other direction. The dragging had not been obvious to me.

And as a second finally, when the dialogue comes up for copying tracks as above, change the "Filename

format" to a more sensible:
%o;%a/%A/%T %t.mp3;%a/%A/%T %t.m4a;%t.wav
as is done in iTunes.

For playing .m4a files, see the comments about xmms under sound.

Internationalization/Accented Characters

The Debian reference manual has a section on "Localization (l10n)". If the system refuses to copy accented characters or Microsoft character encodings from a web page into a terminal window (e.g. running vi in a terminal window, and copying from the web to vi), then this is your problem. As stated there, install the package, locales, or if you have it, `dpkg-reconfigure locales`. Choose the desired country/language localizations, and your default localization. Using `locale -a` then shows you the available localizations. Starting a window with LANG set ("LANG=XXX xterm") then provides the accented characters.

Ethernet:

It worked fine for me with Debian. Debian provides more than one DHCP package, and you must specify a single choice. I'm not sure what the Debian default is, but you will need either the dhcp-client package or the dhcp3-client package (DHCP, version 3). If some things were not initialized well for you at boot time, then try:

```
ifconfig eth0 up
dhclient     [ if you use dynamic TCP/IP addresses ]
```

Also check the entry for eth0 in /etc/network/interfaces . You can also add a line:

OPTIONS='-t 10'

to the case eth0) in the file /etc/dhcpc/config . This tells it to give up searching for Ethernet after 10 seconds, to speed up boot time when you are not connected to Ethernet. Similarly, add a line:

OPTIONS='-t 10 wlan0'

to the case wlan0) in /etc/dhcpc/config .

Wireless:

First, note that on the Averatec, the wireless antenna is only on if you press the "antenna button" near the "power" button. On the Windows side, in the tray of startup icons, the wireless icon reported itself as Broadcom BCM4306 (rev 3). On Linux, more /proc/pci agrees.

NOTE: If you have an RaLink RT2500 wireless (Averatec 3250 and 3270), then try the open source driver at: http://rt2x00.serialmonkey.com/wiki/index.php/Downloads . (Note that the ndiswrapper is reported to have problems with RT2500. The native driver is preferred for RT2500.)

If you have Broadcom, then read on for how to install a driver with ndiswrapper. (If you have Broadcom, a good alternative to ndiswrapper is to go to http://bcm43xx.berlios.de/. Note that that page now recommends going to a newer page at linuxwireless.org. Linux kernel 2.6.17-rc2 and later now has kernel support, but it also needs the bcm43xx-fwcutter tool, explained at that page. They also report a Devicescape 802.11 stack solution, valid for Linux kernel 2.6.18 and later.)

ndiswrapper is available at: http://ndiswrapper.sourceforge.net/ . Note that as a prerequisitie to building it you will need both the kernel source and the package wireless-tools. After the build, you will need the Windows driver for the wireless chip.

If you have the Broadcom (Averatec 3220), Averatec's driver is at: http://www.averatec.com/customercare/3200dl.htm Alternatively, http://ndiswrapper.sourceforge.net/wiki/index.php/List has a list. Look for Broadcom BCM4306 (rev 3). The story seems to be that bcmwl5a.inf is the right driver, but _not_ bcmwl5.inf . (Note that it's bcmw "ell" and not bcmw "one".)

```
Next, I did (assuming ndiswrapper, but note newer, better options, above):
  [ Read INSTALL file in distribution ]
  make install
  cd to directory with Broadcom driver.  Make sure you have both
```

```
                                        bcmwl5a.inf and bcmwl5.sys
ndiswrapper -i bcmwl5a.inf  [install it]
modprobe ndiswrapper      [make it active]
 [add ndiswrapper it to /etc/modules to make this available after every boot]
lsmod | grep ndiswrapper  [to make sure it's there]
ndiswrapper -m            [to make it available the next time you boot]
```

This ndiswrapper required a Linux kernel 2.4.20 or higher or 2.6.0 or higher.

1. **Using your wireless interface to connect:** Next, you will need the commands `iwconfig` and `dhcpcd`. http://www.debian.org/distrib/packages#search_packages shows that iwconfig is in the wireless-tools package, and dhcpcd is in the dhcpcd package. I also installed the dhcp3-client package.

   After this, iwconfig shows a wlan0 interface. You can test it with:

   ```
   iwconfig | grep wlan0
   ```

   Note also: iwlist scan [Must be run as root]

2. **Set the SSID:** By default, your ssid is set to 'any'. At some sites, they may not allow you to connect unless you change it to their SSID. Also,

   ```
   iwconfig wlan0 essid SSID  [ Replace SSID by one for your site. ]
   iwconfig wlan0 channel CHANNEL  [ Replace CHANNEL by channel of
             strongest of access points from previous scan ]
   iwconfig wlan0  [ Check that the ESSID field is your SSID.
             Also check that "Access Point:" has a MAC/Ethernet
             number, and does not say "Not-Associated" ]
   ```

3. **Obtain a dynamic IP address:** Then start it up:

   ```
   dhclient wlan0
   OR:  dhclient3 wlan0
   ```

   If you succeed, you will see something like:

   ```
   DHCPOFFER from 172.16.6.1
   DHCPREQUEST on wlan0 to 255.255.255.255 port 67
   DHCPACK from 172.16.6.1
   bound to 172.16.6.110 -- renewal in 291 seconds.
   ```

4. Run a web browser, if your wireless provider requires you to login with username and password before gaining full access.
5. If dhclient did not succeed, or if you were not able to login to your wireless provider, then run `iwlist wlan0 scan` again. Check if the channel you used corresponds to the channel given by "iwlist wlan0 scan" for the "Access Point" listed by "iwconfig wlan0". Try

   ```
   ifconfig wlan0 down
   iwconfig wlan0 channel CHANNEL essid SSID
   ifconfig wlan0 up
   dhclient3 wlan0
   ```

   but for the correct channel for that access point. Then login to wireless provider as before.
6. **Make the process easily available via ifup:** Add wlan0 to `/etc/network/interfaces` in order to make it work via `ifup wlan0` and `ifdown wlan0` instead of having to use dhcpcd. Additionally, the Debian package `netenv` will automatically start up your network at boot time for you.

   After adding it, I found I had to add a line:

   ```
   OPTIONS='-t 5'
   ```

   to the case `wlan0)` in the file `/etc/dhcpc/config` Otherwise, the search for wlan0 did not timeout, and netenv would hang when called at boot time. (To track this down, I went from "man netenv" to "man interfaces" to "man dhcpcd" to "man dhcpcd-bin". man dhcpcd makes reference

to /etc/dhcpc/config and man dhcpcd-bin makes reference to the timeout option "-t 5".)

Modem:

The modem is SmartLink. To verify this, check the system hardware menu under Windows. (For some reason, `lspci` under Linux reports it as AC97 Modem as indicated below, but the SmartLink driver works.)

```
0000:00:11.6 Communication controller: VIA Technologies, Inc.
                Intel 537 [AC97 Modem] (rev 80)
```

The vendor of the SmartLink modem no longer supports a Linux driver. In the case of kernel 2.6.12, my kernel already had the open source ALSA version of the kernel module installed as a module: "Device Drivers-> Sound-> Advanced Linux Sound Architecture-> PCI Devices-> CONFIG_SND_VIA82XX_MODEM" ("VIA 82C686A/B, 8233 based Modem") The "help" for that module says "support for the MC97 modem on motherboards with VIA chipsets". I do not need to add slamr or snd_via82xx_modem to /etc/modules. lsmod reports that only snd_via82xx_modem is installed. I find that this modem creates the device /dev/pts/0, just as slamr would normally do. I run slmodemd as before, and it creates a symbolic link, /dev/ttySL0. As before, I set a symbolic link, /dev/modem, to /dev/ttySL0.

Apparently, the above solution uses the via82xx_modem kernel module, which presumably does not use the SmartLink modem. Unfortunately, it is not possible to use sound while using the slmodem software in alsa mode.

Luckily, slmodem-2.9.11-*.tar.gz from http://linmodems.technion.ac.il/packages/smartlink/ at the linmodems site *does* work. As of this writing, their latest version is slmodem-20060727.tar.gz .

Once your driver is working, if you have problems with dropped connections, check out their ATCommands.pdf reference.

```
[ You will need the kernel source if you are using a 2.6 kernel.
  I had to get the kernel-source-2.6.*.deb package.  Also, I had to
    build and use the new kernel.  It was not sufficient simply to
    point the slmodem Makefile at the new kernel source.
    ( If you have to build a new kernel, type 'make oldconfig' to
      copy the configuration from your current kernel, `uname -r`.
      In fact, it copies /boot/config-`uname -r` to .config in
      the source distribution.  Then follow the kernel build instructions. )
  Assuming a 2.6 kernel, in the Makefile of that distribution,
      adjust KERNEL_DIR as:
        KERNEL_DIR=/path/to/kernel-source
  It seems to be enough to do 'make' in the KERNEL_DIR (you do not
      need 'make modules'), and then to do 'make' in the slmodem directory.
]
make
# Create driver as device ttySL0
make install
# /lib/modules/2.6.8/extra/{slamr.ko,slusb.ko} should appear
#TRY executing "scripts/debian/slmodemd start"; /dev/ttySL0 should then appear
# A lot of applications look for modem as /dev/modem
ln -s /dev/ttySL0 /dev/modem
#To test on a one-time basis:
#    modprobe slamr;  scripts/debian/slmodemd start;
#    minicom (as root, and set serial port to /dev/modem; then try:
#             ath0        atdt PHONE_NUM   )
# Type slmodemd each session before using the modem, or else automate as
#  part of single-user mode startup, which precedes rest of initialization:
cp scripts/debian/slmodemd /etc/init.d/slmodemd
ln -s ../init.d/slmodemd /etc/rcS.d/S90slmodemd
# The slmodemd script should now start automatically at boot time.
# It then starts the daemon that creates /dev/ttySL0
# You can also start the daemon manually with:  /etc/init.d/slmodemd start
```

An easy way to test the modem is to go to minicom, and type something like `ath0'. If the modem is functioning, you should see the characters echoed. To get minicom working, you will need to configure

it (under root) so that the serial device is /dev/modem .

So far, I've found the modem driver to be stable under 2.6 Linux kernel.

To set up PPP, I used pppconfig, which was fairly painless Because of call waiting, any new incoming phone call would interrupt my session. So, I added "*70," as a prefix to the phone number. After that, pon, poff and plog are available. Note that by default, root is not allowed to use pon (see /etc/ppp /pap-secrets). *So, you must test as an ordinary user.* I also did `chmod o+r /var/log/syslog` so that plog would work for an ordinary user. I don't log any passwords to syslog. With some techniques (but not with pon), you might have to add a user to the group, dip, in order to use pon, since pppd is in that group.

CD-RW/DVD (writing CD disks):

For some reason, when I installed Debian 3.0, /dev/cdrom was a symbolic link to /dev/scd0 (SCSI disk). As `dmesg' shows, it should be a symbolic link to /dev/hdc. `rm /dev/cdrom; ln -s hdc /dev/cdrom` fixed this. The Debian reference manual says that writing CDs should work fine with a 2.6 kernel. For the use of cdrecord, it is best to have a kernel 2.6.9 or later. I currently use cdrecord version 4:2.01+01a03-5.

2.6.8 kernel only: If you have the older 2.6,8 kernel, do:

```
strace cdrecord -eject -v dev=/dev/hdc \
-dao driveropts=burnfree -data IMAGE.iso \
2>&1 | grep -v ^gettimeofday | grep -v ^ioctl
```

(Be prepared for lots of extra printout. Possibly, insert the option `-dummy` to turn off the laser if you only want to test.) Note that -data is for data CDs and -audio exists for audio CDs. See the cdrecord man page for details.

For later 2.6 kernels, the command bwlow should work, with SCSI_ADDR being your virtual SCSI address.

```
nice --10 cdrecord -v -eject speed=2 dev=SCSI_ADDR disk.img
```

To discover your virtual SCSI address, the package cdrtools-doc has a file /usr/share/doc/cdrecord /README.ATAPI.setup, which says to do: cdrecord dev=ATA: -scanbus This doesn't work for me, but I succeed with: cdrecord dev=/dev/hdc -scanbus This should produce the SCSI_ADDR after 'scsidev:'.

```
scsibus1:
        1,0,0   100) 'QSI     ' 'CDRW/DVD SBW-242' 'UX02' Removable CD-ROM
```

An alternative using cdrdao is:

```
cdrdao write --device /dev/hdc --speed 4 cdrdao.toc
[ where I want to write disk.img and cdrdao.toc contains: ]
CD_ROM

// Track 1
TRACK MODE1
COPY
DATAFILE "disk.img"
```

For CD-RW:

```
nice --10 cdrecord -v -eject blank=fast speed=2 dev=/dev/hdc disk.img
```

OR:

```
nice --10 cdrecord -v blank=all speed=2 dev=/dev/hdc disk.img
nice --10 cdrecord -v -eject speed=2 dev=/dev/hdc disk.img
```

The "nice --10" is to run at high priority, since any pauses will ruin the recording. To make cdrecord usable for non-root users, do the following, and then "adduser USER cdrom" or directly modify

/etc/group .

```
dpkg-reconfigure cdrecord
```

To copy a directory to a CD, convert the files in TARGET_DIRECTORY into an ISO image for writing to a CD, do:

```
mkisofs -r -V volume_id -J -T -o disk.img TARGET_DIRECTORY/
```

In order to inspect an ISO image before writing to CD, make sure that /cdrom is not currently mounted and do:

```
mount -t iso9660 -o ro,loop disk.img /cdrom
cd /cdrom  # and look around
umount /cdrom   # after cd'ing out of /cdrom
```

Alternatively, a simpler method, that also allows you to modify the ISO image is:

```
modprobe loop    [to make sure kernel module, loop, is available]
mount -t loop disk.img /cdrom
cd /cdrom  # and look around
rm -rf newcd; mkdir newcd
rsync -a -H --exclude=ANY_FILES /cdrom newcd
umount /cdrom   # after cd'ing out of /cdrom
mkisofs ... newcd/
```

The Debian reference manual also shows how to make disk.img bootable. Do:

```
mkisofs -b REL_PATH/TO/BOOT_IMAGE -c REL_PATH/FOR/BOOT.CAT -no-emul-boot
```

where boot.cat doesn't exist (mkisofs will create boot.cat, but it may not overwrite an existing one??) and REL_PATH/TO/BOOT_IMAGE and REL_PATH/FOR/BOOT.CAT are relative paths to TARGET_DIR.

```
[ Note that the Debian reference manual omits  -no-emul-boot .
  But 'man mkisofs' says that the boot image must then be a
  floppy image. ]
If your bootable CD is based on Knoppix, you probably want the
     following example:
mkisofs -b TARGET_DIR/boot/isolinux/isolinux.bin \
     -c TARGET_DIR/boot/isolinux/boot.cat -no-emul-boot ...
```

I'm guessing that the name isolinux refers to the idea of creating bootable ISO images in Linux. isolinux.bin is probably the program with the initial splash screen asking you what kernel/options to choose.

In order to convert a data CD to an ISO image on disk, do (as root with Linux 2.6 kernel):

```
readcd dev=/dev/cdrom
```

and choose option "11:read disk" (at least as tested with readcd version 2.01). The readcd command is part of the cdrecord Debian package. You will then be guided interactively. Note that "block size" and "sector size" are synonymous in this context.

To backup an audio CD, the Debian Reference Manual recommends:

```
# apt-get install cdrdao #disk at once
# cdrdao read-cd --device /dev/cdrom --paranoia-mode 3 my_cd # read cd
# cdrdao write --eject --device /dev/cdrom --speed 8 my_cd # write new CD

cdrdao does a real copy (no gaps, etc...).
```

Note that when you buy blank CDs, there are "data CDs" and "music CDs". If your CD player doesn't say that it can play CD-R media, you may need to use slightly higher priced "music CD" media.

I haven't tried burning a DVD, but here are some pointers:

- The Debian Reference Manual has a subsection on DVDs under the CD section.
- Linux Magazine -- DVD Burning, DVD writers, ...
- How to burn a DVD-Video under Linux with mkisofs and dvdrecord (which refers to dvdrtools).
- K3b -- the CD Kreator (optimized for KDE, claimed to be very easy to use)
- dvdrip

Note also DVD Shrink, which is free software for backing up your DVDs under Windows. One use for this is to make a region-free backup, in case your DVD is set for the wrong region for your DVD player. DVD Shrink doesn't appear to be open source.

RAM Disk:

RAM disks lose their files on shutdown or power failure. With that caveat, verify that /dev/shm exists. To create the RAM disk, choose a suitable location and mount it. For example:

```
mkdir /media/tmpfs
mount -t tmpfs -o size=1G,nr_inodes=5k,mode=700 tmpfs /media/tmpfs
```

Additionally, there is a /dev/shm device that the operating system uses for the POSIX shared memory abstraction. It can be resized via:

```
mount -o remount,size=2G /dev/shm
```

An older method to create a ramdisk without using tmpfs is:

```
# To make a ramdisk of 2 MB (2048K):
mke2fs -vm0 /dev/ram 2048
mkdir /mnt/ramdisk
mount /dev/ram /mnt/ramdisk
# Then write some script that will copy your important files
#   to hard disk every 10 minutes or so.
```

Following the same principles, you can look inside an existing CD or partition image:

```
mkdir /mnt/cdrom-extra
mount -t iso9660 -o ro,loop CD-IMAGE.iso /mnt/cdrom-extra
```

Touchpad and Keyboard:

The Synaptics touchpad works fine. The Debian reference manual says to load the package tpconfig if you want to configure the touchpad options. However, when I do: `tpconfig -i` as root for tpconfig version~ 3.1.3, I get:

```
auditor:/usr/local/jdk1.5.0/docs/api/java/lang# tpconfig -i
fatal:
No Synaptics or ALPS touchpad device found
```

The keyboard also works fine. In Gnome 2.10, "Desktop->Preferences->Keyboard ->Layout Options" has options to create a "compose" key for accented characters and international characters (e.g. Yen), but it doesn't work for me, yet. The Gnome Keyboard Indicator applet (different national keyboards, incl. U.S. w/ dead keys), and Chacter Palette applet (international characters can be copied and pasted) may also do the trick for you.

Power management and Adjust CPU Speed:

```
I added
  battery
to /etc/modules.
acpi -b          now works
and   /proc/acpi/battery   now exists.
```

It shows battery level. Some other features, such as suspend, may depend on later 2.6 kernels?

The Gnome CPU Frequency Scaling Monitor shows you if you have frequency scaling. This requires that it be configured in the kernel. (I think it's the default for Debian.) It also requires powernowd, below.

Finally, with the 2.6.12 kernel, Etch provides packages *hibernate*, powernowd, powertweak, lspowertweak, and laptop-mode. The packages `noflushd` (spin down hard disk) and `powernowd` (adjust speed of CPU) may be useful for power conservation. powernowd requires the CPUFreq kernel module.

Laptop mode is described in: `/usr/src/linux-source-2.6.12/Documentation/laptop-mode.txt` (requires linux-source-2.6.12 package). It says that `echo 5 > /proc/sys/vm/laptop_mode` will turn it on with a reasonable value, while 0 turns it off. It also notes that hard disks in desktop computers typically have *a very limited lifetime measured in spindown cycles (typically about 50,000 times). Check your drive's rating, and don't wear down your drive's lifetime if you don't need to.*

Hibernate/Suspend:

I haven't tried Hibernate/Suspend, but note the existence of the Debian package *hibernate*. Currently, with the kernel 2.6.12, it says that it needs the "Suspend 2" patch to the kernel. Gordon Cormack reports success for *hibernate* by adding "resume=/dev/hdaX" to the line in "/boot/grub/menu.list", where `/dev/hdaX` is the *swap* partition. He also reports that he needs to stop the wireless and unload the wireless module using `modprobe -r` *before* hibernating. To hibernate, he switches to a non-windows console and types `echo 4 > /proc/acpi/sleep`. (<CTRL><ALT><F1> or `chvt 1` to switch to console 1. The original graphics console is back at number 7 (F7).) See Cormack's page for details.

Others recommend the swsuspend2 patch to the 2.6.11 kernel.

USB (mice, keyboards, flash drives, digital camera, etc.)

For USB, ideally, things should just work. In case they don't, here's a checklist to troubleshoot USB Flash drives and other USB storage. Please note that devfs and udev are now obsolete for Debian. If you are using these, please upgrade devfs->udev and hotplug->hal. (udev may require an upgrade to a more recent kernel, 2.6.12 or greater.)

1. Make sure you have the usb-storage kernel module present:

```
lsmod |grep usb_storage
[ If not present, add   usb-storage   to /etc/modules, and reboot ]
```

2. The next layers needed in the protocol stack are: udev, hal, hal-device-manager, gnome-volume-manager. *(What is hal-device-manager needed for? Where do dbus and avahi-daemon fit?)*
3. Udev recognizes the kernel events and will install devices. (For example, when a USB stick is inserted, it is installed for me as the new device /dev/sda1. "dmesg|tail" shows that /dev/sda1 was created.)
   With kernel 2.6.17, I could no longer use my USB stick, and I instead saw messages like:

```
usb 1-2: new high speed USB device using ehci_hcd and address 2
Can't accept address 2, error -101
```

   According to Ubuntu Bug #49367, this is a bug with the "experimental individual usb suspend", and it should eventually be fixed in a newer kernel. In the meantime, they recommend to disable USB 2.0 via
   "modprobe -r ehci_hcd" and then reinsert the USB stick. You can permanently disable it via:
   "echo "blacklist ehci_hcd" > /etc/modprobe.d/blacklist-ehci",
   but don't forget to re-enable it with a newer kernel that is hopefully free of this bug.
4. Hal can modify where those devices are mounted. (For example, when a USB stick is inserted, it is mounted at /media/usbstick for me, as seen by "mount".).
5. Gnome-volume-manager places an icon on your desktop to reflect the new removable media. (If not using gnome, then you will need an alternative package for this functionality.)

A USB flash drive may mount automatically using udev/hal (Hardware Abstraction Layer). (Also, don't forget to unmount before pulling it out with `umount`, or you could corrupt its filesystem, and have to

remake it.) To mount the usb flash drive manually, I did:

```
usbview  # Verify USB drive is recognized.  Requires usbview package.
         # Alternatively: cat /proc/bus/usb/devices
dmesg | tail  # See if udev recognized USB key insertion with a device
mkdir /mnt/usbdisk
# YOU MAY NEED /dev/sda OR /dev/sda1 OR /dev/sdb OR /dev/sdb1, ETC.
mount /dev/sda1 /mnt/usbdisk
# AFTER USING IT, DON'T FORGET:
umount /mnt/usbdisk
```

If you pull out the USB disk without first unmounting, you risk corrupting the files, and having to reformat the USB disk. Note that some sites recommend `mount /dev/sda1 /mnt/usbdisk` (first partition of scsi disk a). Try that if /dev/sda doesn't work. Also, try /dev/sdb, /dev/sdb1, for drive b, etc.

A digital camera will show up as one or more of (i) a USB storage device, (ii) a device using PTP (Picture Transport Protocol), or (iii) a proprietary interface. The URL http://www.wlug.org.nz/DigitalCameraSetup has excellent information. These days most vendors provide PTP, and many provide USB storage device interface. If a USB storage device is supported, you can simply copy the photos from the virtual disk. For example our Nikon Coolpix 3200 seems to support USB storage, but I couldn't get PTP to work (neither "Nikon Coolpix 3200 PTP" nor "USB PTP Class Camera"), while our Canon A5100 worked for us only using "USB PTP Class Camera".

The text-oriented Debian package *gphoto2* handles PTP devices and vendor-specific interfaces. To use it, add all users to the camera group in /etc/group, so that the /etc/hotplug/usb/libgphoto2 script will give permission to those users for the corresponding USB device. Then, to import files, do (for example) `gphoto2 --port "usb:" --camera "Kodak CX4230" --get-all-files`, although a simple `gphoto2 --get-all-files` or `gphoto2 -P` should also work. (See `man gphoto` for other options, such as `--delete-all-files` for reformatting the camera flash memory.) The file `/usr/share/doc/libgphoto2-2/README.Debian` has further information.

Additionally, to view and organize photos on the computer, the Debian *gthumb* package provides a nice GUI. It installs a Debian menu item "Applications->Graphics->gThumb Image Viewer". The gthumb application has a menu item "File->Import Photos". If you import from your camera with gthumb, instead of using gphoto, then attach and turn on your camera, click the camera icon of the "Import Photos" dialog, and choose the camera model "USB PTP Class Camera" from the pull-down menu in the "Camera Model" dialog box. The specific camera models do not work for me.

A USB mouse did not work at first. In using a USB mouse, I got a message: usb.agent: usbmouse: blacklisted in some logs. (Try `grep -i usb /var/log/*`) Apparently, `/etc/hotplug/blacklist` contains:

```
# these drivers are very simple, the HID drivers are usually preferred
usbmouse
usbkbd
```

After commenting out usbmouse and rebooting, the USB mouse worked. I assume the same would hold for a USB keyboard.

For a Logitech USB QuickCam IM videocam, I saw this page, which seemed to have specific advice for USB device 46d_8a6 (my device), and calls it QuickCam IM/Connect. They say to install the spca5xx-YYY Debian package, a kernel module that needs to correspond a. exactly to your kernel. (Check the kernel name with `uname -a`.) They also write that the the newer versions of the driver are no longer called spca5xx but gspcav1 v 1.00.11 is the latest at the time of writing, again, see the changelog.) Don't forget `modprobe spca5xx` to install the module. Also, see the list of recognized cams, although some other cams also work.

If you succeed in installing the module, you should see `/dev/video0`. Then, don't forget

```
adduser USER video
```

Next, check out some utilities that use video4linux (v4l and v4l2).

If that doesn't work, try this other possibility that I found. It didn't work for my model, but it may for your model. For a Logitech USB webcam/videocam, I followed instructions at: Debian Manual --- Setting up a Logitech Webcam. Fundamentally, the instructions were to get the package qc-usb-source, and follow the instructions. The script within it, ./quickcam.sh, is well-written and leads you through the installation.

./quickcam.sh said it didn't recognize my videocam (too new?). I continued to press in the script until the end. I then entered as root:

```
depmod -a
modules/qc-usb-source# modprobe quickcam
```

Other experiences with USB devices are welcome. Until then, here are some random links from the web.

- FAQ
- Linux and USB 2.0
- USB Devices on Linux
- USB Digital Camera HOWTO
- Making USB Storage Devices work seamlessly under Linux (refers to a small script to use)

Reconfiguring the kernel:

Knoppix configures the kernel to include most features, including a variety of sound drivers to fit many computers. Debian by default configures the kernel to include no sound drivers at all (under the assumption that you will reconfigure the kernel??). Some other kernel features may similarly be missing.

If you do build a new kernel, I recommend having Knoppix or some other Linux rescue CD that will allow you to fix things in case there is a problem with your new kernel. (See Recovering when a new Debian kernel doesn't boot for boot problems.) Also, keep a copy of the old kernel and modules. The normal kernel configuration tends to create an old copy for you automatically. Here's what I did to build a new kernel, after getting the Debian package kernel-source-*.deb:

For an excellent overall description, see the new Kernel-HOWTO (currently, then click on Kwan Lowe's *Kernel Rebuild Guide*). I recommend to read that document carefully which clearly explains all issues except some of the configuration options. For configuration, also check out the first two links below for configuring nVidia video drivers, USB flash drivers, etc. Here are other good descriptions of compiling the kernel from the web, which also cover additional issues.

- Debian Kernel-Compile Howto (Kernel 2.6) (very up-to-date, with mkinitrd, issues of USB flash drives and nVidia video)
- Andrew's Debian nVidia HOWTO (If you have nVidia video hardware, you want this for added performance.)
- The Very Verbose Guide to Updating and Compiling Your Debian Kernel (good advice about configuration)
- Creating Custom kernels with Debian's kernel package system
- Compile a Linux kernel especially under Debian
- Debian Kernel Compile Howto (official Debian way, good advice on configuring)

Before starting, here are some general pieces of knowledge to keep in mind:

- If you use the ext3 filesystem, configure the Debian kernel with ext3 builtin (_not_ as a module) and IDE and IDE_BLK_DEV also builtin (not as modules). If they are modules, your builtin kernel will not know how to access your IDE hard disk, or mount the ext3 filesystem on hard disk. You will need the extra complication of using *initrd* (initial RamDisk), which will start your

builtin kernel in RAM, and will then load the ext3 module before trying to mount an ext3 filesystem on hard disk. You may also need to add ext3 and the modules for your ide disk to /etc/modules to force them to autoload, but I'm not sure about that.

- Roughly, you will need to do:
  (0) set EXTRAVERSION=-20050125 (current date; or whatever else you prefer); This gives your kernel a unique name. Now there is no name conflict between files for your new and old kernel. original kernel version
  (1a) make clean (to remove any old object files)
  (1b) make mrproper (optional; to remove old configuration files, including to remove .config)
  (2a) make oldconfig (to take your old kernel configuration as a default; If .config exists, it uses that; if /boot/config-VERSION for current kernel exists, that is copied into .config)
  (2b) make menuconfig (to configure the kernel; has a prereq of the libncurses5-dev and libncurses5 packages --- not part of the first three Debian disks for Sarge; Don't forget to make ext3 *builtin* and not a *module*, as explained above.)
  (3) make (to compile the kernel)
  (4) make modules (to compile the modules)
  (5) make modules_install (install the modules by copying them to /lib/modules/VERSION/
  (6) make install (should copy arch/i386/boot/bzImage to /boot/vmlinuz-VERSION; and copy System.map to /boot/System.map-VERSION; Note that System.map is unrelated to /boot/map, a file used by lilo) (7) Hopefully, you had your ext3 or other filesystem drivers built in. (If you need it at boot time, and you insisted on making it a module, you will need to use mkinitrd to create an initrd. Then copy it to /boot/initrd-VERSION.) (7) Edit /etc/lilo.conf (or /boot/grub/menu.lst if you use GrUB). For lilo, make sure that vmlinuz, map (for System.map), and initrd (if you didn't follow advice) all point to the right location. Often, they are set to symbolic links, and you must track down the target.) Make sure that root points to the correct partition. Keep a 'image=/boot /vmlinuz.old' stanza pointing to your old kernel, in case the new one doesn't boot. (8) If you have more than one partition with Linux, mount the other partitions, and then edit the additional stanzas of /etc/lilo.conf to show (for example) 'image=/mnt/hda6/boot/vmlinuz', 'root=/dev/hda6', 'initrd=/mnt/hda6/boot/initrd', 'label=OtherLinux' (9) /sbin/lilo (to permanently install the new boot options) (10) Reboot. If there is an error message, pressing ScrollLock with pause the boot process for you to read it. If the computer freezes, you can hold down the four keys: <CTRL> <ALT><SysRq>b to force a reboot (usually). If already under Gnome, try <CTRL><ALT><1> followed by <CTRL><ALT><SysRq>h (for help). Then do <CTRL><ALT><7> to return to the Gnome GUI. If you succeed in booting, 'dmesg' will often describe problems in loading an individual modules for that session. '/var/log/syslog' will contain error messages from that session and from previous ones.

If you've come this far, you should probably go back and read the information above about compiling the kernel. Below are my own notes, which are not as good. But here they are, anyway.

```
INSTRUCTIONS FOR 2.6 KERNEL
                I still make kernels the original, non-Debian way.
                For the earlier Debian kernels, I had to turn on Alsa and
                  CPU Frequency scaling
                cp /boot/config-2.6.8 /boot/config-2.6.8.orig

                [ Some people call this directory  /usr/src/linux-2.6.8 ]
                cd /usr/src/kernel-source-2.6.8
                [ If you have no other backup kernel, besides a 2.6.8 image,
                  I suggest editing Makefile, to add a unique string for
                  EXTRAVERSION= ]
                more README
                make mrproper    [Clean out old .o files]
                [ Next, type 'make menuconfig'
                   (libncurses5 and libncurses5-dev packages needed),
                   and select: "load from file"
                   /boot/config-2.6.8 if it exists (to see old choices);
                   (A shortcut is to have first typed 'make oldconfig' to
                    copy /boot/config-`uname -r` to .config .)
                   Then modify the options as desired and save. ]
                make menuconfig
                [ Next, maybe record your choices; Does make install do this?
                  cp .config /boot/config-2.6.8-$EXTRAVERSION ]
```

```
make
make modules_install install
grep bzImage README
# I already had a backup kernel image.  Make sure you're
#   not clobbering your only backup kernel before doing
#   the next step.  If you added to Makefile:EXTRAVERSION= ,
#   then you should add that to the 2.6.8 below.
#This next `cp' seems to be done already by Makefile .
#The README says you need to do it, but you don't.
# cp arch/i386/boot/bzImage /boot/vmlinuz-2.6.8
ls -l /vmlinuz
[ Do something like:
  rm /vmlinuz
  ln -s /boot/vmlinuz-2.6.8-$EXTRAVERSION /vmlinuz ]
[ I use the older LILO.  You will need to adjust
     the menu.lst of the newer GRUB, if you use it.]
[ Edit /etc/lilo.conf to make sure you have an entry for
      /vmlinuz
   similar to the existing entry in /etc/lilo.conf
  Also, make sure you have an entry /vmlinuz.old that
      points to a previous kernel, in case
      the new one doesn't boot. ]
Check /etc/lilo.conf, and edit it if necessary.
lilo  [ Do this during current session; Else system might
          be unbootable when you reboot ]
[ Reboot, and verify with `/sbin/lsmod' that you have your
  desired modules. ]
```

Installing a kernel module:

You can install a new kernel module without having to compile the kernel source. To do so, do:

```
cd /usr/src/linux-source-XXX
make mrproper   # Erase all objects, binaries, temporary files, .config
cp /boot/config-XXX .config
make oldconfig
make modules_prepare
```

Recovering when Debian kernel doesn't boot:

When updating to a new kernel through the Debian package system, or reconfiguring and adding a new kernel, it seems to be too easy to make Debian unbootable (at least it's easy to do for me personally :-)). Luckily, it is not too hard to fix, using Knoppix or some other LiveCD. There are several techniques, but here is mine. This is based on the lder LILO, which I use. Some people use GRUB.

```
Reboot under Knoppix (or with knoppix26 to fix a 2.6 kernel).
Open a terminal window.
su root
# Note that "-o dev" is important in order to execute lilo
mount -o dev /mnt/hda6  # assuming /dev/hda6 is your root partition
chroot /mnt/hda6
# You are now root, looking at your old root partition, and you will be
#  using your old binaries.  The only difference is that you are
#  running the Knoppix kernel instead of your old one.
```

Then decide what you did wrong. If you simply forgot to type lilo, you can do it now, and reboot normally. If it is more subtle, review what lilo is doing for you. For each O/S or kernel, lilo needs to know:

1. on which partition is the kernel (specified by "root=" parameter, typically something like root=/dev/hda6; if on a different Linux partition, you might need, for example, "root=/dev/hda7" in the section for that kernel image.)
2. where is the image in that partition (specified by "image=" parameter, typically something like "image=/vmlinuz"; but if the kernel is on a different Linux partition, then mount the remote partition (e.g.: mount /dev/hda6 /mnt/hda6) and verify that "image=/mnt/hda6/vmlinuz" appears in lilo. Try "ls -l /mnt/hda6/vmlinuz" to verify that the file is currently accessible.)
3. where is the initial ramdisk (assuming the 2-stage boot process) (specified by "initrd=" parameter, typically something like initrd=/initrd.img, but do as above if it's on a different partition.)

The files /vmlinuz and /initrd.img are usually symbolic links. Check what they link to.

Then read the lilo man page carefully. You can compare what the previous lilo invocation did with `lilo -I ...`. If you don't see how to fix it, you can also restore a previous boot sector with `lilo -u`. By default, you get the original boot sector before you installed Debian. But if you previously did `lilo -s`, then you have an intermediate (and hopefully working) boot sector.

Minor bugs that I haven't fixed

These are little bugs for which I find it easier to live with them than track down their cause and fix them. Probably some of the bugs are unique to the way I happened to install and upgrade my system.

- From the login screen, if I change my window manager to KDE, it works fine, but then on logging out (exiting the KDE session), the computer freezes.
- The tpconfig package doesn't work for me (see the touchpad section)
- bb is a great application for demo'ing Linux. For me on this system, it works fine if you choose no music. With music, the screen freezes when the music starts. But it's still an awesome demo. (However, I can still run two bb's at same time: one with music and one without.)